Journal of Systems Architecture 94 (2019) 24–31

Contents lists available at ScienceDirect

Journal of Systems Architecture

journal homepage: www.elsevier.com/locate/sysarc

Bio-inspired cryptosystem with DNA cryptography and neural networks

Sayantani Basu [a], Marimuthu Karuppiah [a,*], Mita Nasipuri [b], Anup Kumar Halder [b], Niranchana Radhakrishnan [a]

[a] School of Computer Science and Engineering, Vellore Institute of Technology(VIT), Vellore- 632014, Tamil Nadu, India
[b] Department of Computer Science and Engineering, Jadavpur University, Kolkata-700032, India

ARTICLE INFO

Keywords:
Cryptosystem
Bio-inspired
Central dogma
Key generation

ABSTRACT

Bio-Inspired Cryptosystems are a modern form of Cryptography where bio-inspired and machine learning techniques are used for the purpose of securing data. A system has been proposed based on the Central Dogma of Molecular Biology (CDMB) for the Encryption and Decryption Algorithms by simulating the natural processes of Genetic Coding (conversion from binary to DNA bases), Transcription (conversion from DNA to mRNA) and Translation (conversion from mRNA to Protein) as well as the reverse processes to allow for encryption and decryption respectively. All inputs are considered to be in the form of blocks of 16-bits. The final outputs from the blocks can be concatenated to form the final cipher text in the form of protein bases. A Bidirectional Associative Memory Neural Network (BAMNN) has been trained using randomized data for key generation which is capable of saving memory space by remembering and regenerating the sets of keys in a recurrent fashion. The proposed bio-inspired cryptosystem shows competent encryption and decryption times even on large data sizes when compared with existing systems.

1. Introduction

Cryptography is the science of studying techniques of hiding information in order to allow secure communication and exchange of information. The process of applying logical and mathematical approaches to hide data is known as encryption. The encrypted form of data is often referred to as cipher text. The reverse process of retrieving back the original data is known as decryption. The complete system comprising the encryption algorithm, key generation algorithm and decryption algorithm is named as cryptosystem. Currently, developments in the arena of cryptography have been directed toward the principle of confidentiality, integrity and availability [1–9].

Symmetric and asymmetric ciphers have already been designed to provide secure encryption and information hiding [10–12], but at present, there is also an added requirement in designing encryption algorithms capable of providing optimum security and storage. In the near future, storage problems will become a burning issue in the case of conventional cryptographic algorithms. The proposed bio-inspired cryptosystem that uses both machine learning and biological approaches is capable of providing security as well as efficient storage to data, which makes it possible to encrypt and decrypt even large amounts of data.

Bio-inspired cryptography is an emerging approach for information security and storage because of its enormous storage and securing capabilities. DNA molecules are capable of parallelization, efficient utilization of energy, containing high information density and storage [13]. DNA cryptography can encrypt data with huge combinations of four nucleotide bases $\Sigma = \{A, C, T, G\}$ while machine learning can aid in generating keys which are difficult to guess. DNA cryptography is a relevant example of how the natural processes of encoding, transcription and translation originally present in the central dogma can be mimicked for the purpose of cryptography.

In the proposed bio-inspired cryptosystem, DNA cryptography has been used for the encryption and decryption algorithms which try to simulate the processes of genetic encoding, transcription and translation from the Central Dogma of Molecular Biology (CDMB). For the key generation algorithm, a Bidirectional Associative Memory Neural Network (BAMNN) has been trained using randomized datasets for the purpose of generating and storing the keys during the processes of encryption and decryption. This system is capable of providing sufficient level of security while offering enough confusion and diffusion. It also effectively saves memory and reduces the number of cryptographic operations when compared with conventional algorithms in cryptography.

The rest of this paper is organized as follows: Section 2 discusses the literature related to the proposed area of study, Section 3 details the theoretical background, Section 4 explains the proposed methodology, Section 5 discusses the results obtained, Section 6 mentions the performance analysis and Section 7 concludes the paper and suggests future work.

* Corresponding author.
E-mail address: k.marimuthu@vit.ac.in (M. Karuppiah).

https://doi.org/10.1016/j.sysarc.2019.02.005
Received 18 September 2018; Received in revised form 3 February 2019; Accepted 4 February 2019
Available online 5 February 2019
1383-7621/© 2019 Elsevier B.V. All rights reserved.

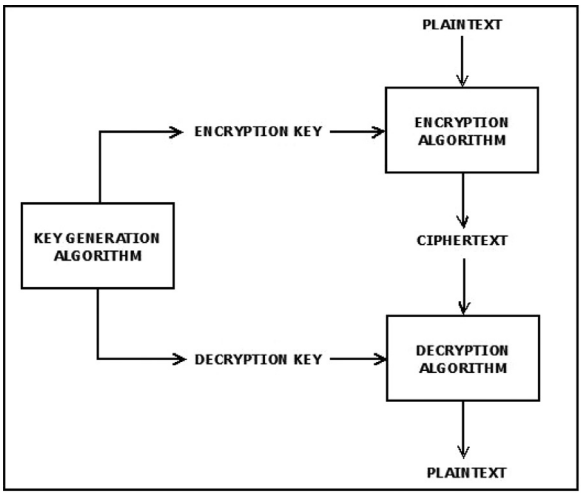

Fig. 1. Components of a cryptosystem.

2. Related work

The past decades have seen an enormous progress in the development of cryptographic algorithms. The initial developments were in the form of monoalphabetic ciphers, polyalphabetic substitution ciphers, transposition ciphers and block ciphers [14]. These were followed by more advanced encryption algorithms such as AES, DES, RSA and SHA [15,16].

However, future demands will expect cryptographic algorithms to be capable of encrypting large sizes of data and memory efficiency, which can be aptly filled by the use of DNA and Machine Learning based approaches in Cryptography [17–20]. Xie et al. [21] have employed crypto-nets, or in other words, neural networks for the purpose of encryption. They have used homomorphic encryption, modified the activation function and trained the neural network for their proposed protocol.

The authors in [22] have applied echo state networks in the field of cryptography. Echo state networks are a form of recurrent neural networks. The authors have proposed a scheme to train a neural network for encryption and secure communication of messages. Sagar and Kumar [23] have proposed a symmetric key method using genetic algorithm for encryption and error back propagation neural network for decryption.

Kalsi et al. [24] have proposed a key generation scheme using Genetic Algorithm and Needle-Wunsch Algorithm. They have performed encryption and decryption using DNA computing based on the processes of transcription, translation, DNA sequencing and Deep Learning. The authors have evaluated the time taken for encryption and decryption based on size of the input. The authors have suggested conjugation with logical operations for a more secure cryptographic system, which has been considered in the present work. Kang Ning [25] has proposed a Pseudo-DNA Cryptography method that simulates the processes of the central dogma of molecular biology for the purpose of encryption and decryption. The theoretical analyses have been performed based on the size of plaintext and key and the time taken (in milliseconds) for encryption and decryption. Providing better confusion has been suggested as well as prevention against the differential cryptanalysis attack.

MingXin et al. [26] have proposed a DNA symmetric-key cryptosystem (DNASC). They have performed encryption and decryption using DNA probes and embedded the cipher text in a microarray. They have evaluated the security and parallelism of their system. However, the security of their system relies only on computational properties which are not provably secure. Hussain [27] have suggested a hybrid DNA approach to cryptography involving Data Encryption Standard (DES) and central dogma of molecular biology. They have shown that their proposed system is better than conventional DES with respect to parameters such as input data limit and attack resistance.

Jogdand and Bisalapur [28] have proposed a key generation algorithm based on neural network. They have evaluated the performance of their proposed algorithm with regard to security, synchronization time and randomness. Randomization is also performed in the present work while performing key generation. They have proposed carrying out secure distribution of the generated keys in the future. The authors in [29] have proposed a Hopfield Network Based Neural Key Generation (HNBNKG) for encryption, key generation and decryption. Their technique ensures that instead of sharing of the secret key, the weight vectors of the trained network can be shared between the sender and the receiver to reduce attacks. This has also been ensured in the present work.

Hence, in the present work, a bio-inspired cryptosystem has been proposed to overcome the shortcomings of previously proposed systems.

3. Theoretical background

3.1. Cryptosystem

Cryptography has become an integral part of securing any system at present as well as protecting important data and information against attacks. Any modern cryptosystem generally consists of the following components and algorithms [1]:

(i) Plaintext: The original readable text to be converted to cipher text is known as the plaintext.
(ii) Cipher text: The text obtained after encryption is known as the cipher text.
(iii) Key: A string of characters generated using the key generation algorithm. A symmetric key cryptosystem is one in which the same key is used for encryption and decryption whereas the asymmetric key cryptosystem is one which makes use of public and private keys.
(iv) Encryption Algorithm: This algorithm is used for converting plaintext to cipher text using the key.
(v) Decryption Algorithm: This algorithm is used for converting cipher text to plaintext using the key.
(vi) Key Generation Algorithm: This algorithm is used for generation of keys. This is an integral part of the cryptosystem as the generated keys are used in combination with the plaintext in order to produce the cipher text. Fig. 1 shows the components of a cryptosystem.

3.2. DNA Cryptography and central dogma of molecular biology

DNA, which stands for deoxyribonucleic acid, is a molecule in living organisms that stores genetic information. It is also responsible for carrying hereditary material in the form of genes. A nucleotide is the basic building block of a DNA molecule. The nucleotide comprises of a nitrogenous base, deoxyribose sugar and at least one phosphate group. The various nitrogenous bases are Adenine (A), Cytosine (C), Thymine (T), and Guanine (G). Various combinations of these bases are capable of storing information. RNA (ribonucleic acid) contains the same bases as DNA except Thymine (T) which is replaced by Uracil (U).

Since the DNA strings are represented using the alphabet $\Sigma = \{A, C, T, G\}$ in contrast with the binary strings which are represented using the alphabet $\Sigma = \{0, 1\}$, so DNA strings are capable of storing much more information compared to binary strings. In the present work, the binary strings 00, 01, 10, 11 are used to denote A, C, G and T respectively. In addition to storing large quantities of data, DNA molecules are also able to store information in a compact form.

Adleman [30] first proposed an application of DNA computation in the area of cryptography. DNA cryptography provides a challenge to conventional cryptographic algorithms in both theoretical and

technological aspects. The components of DNA cryptography used in the proposed cryptosystem are as follows:

DNA sequencing: The process by which the sequence of bases is determined in DNA is known as DNA Sequencing.

DNA encoding: The process by which any given readable plaintext is converted into a sequence of DNA bases is known as DNA encoding. In this context, the user input has been converted into binary by using probability-based Huffman encoding [31]. The binary coded string is then XORed with the key and converted into the corresponding genetic code ($00 \rightarrow A$, $01 \rightarrow C$, $10 \rightarrow G$ and $11 \rightarrow T$).

The central dogma of molecular biology (CDMB) and the structure of DNA were proposed by Watson and Crick [32,33]. CDMB states the residue-by-residue transfer of sequential information. CDMB also gave an insight into the biological processes of transcription and translation which are discussed as follows:

(i) Transcription: When an enzyme RNA polymerase attaches itself to the strand of template DNA, it catalyzes the production of mRNA (messenger RNA). This process is known as transcription.
(ii) Translation: When ribosomes get attached to the mRNA strand produced from transcription, it gets converted into a string of amino acids (units of protein). This process is known as translation.

The processes of reverse transcription and reverse translation may not be feasible in the context of molecular biology. For the sake of the proposed method, all of the above-mentioned processes have been algorithmically simulated.

3.3. Machine learning and associative memory

Earlier, machines had to be programmed in order to execute any particular task. With the advent of artificial intelligence, machines can be trained to learn and think based on situations in real life. Machine learning is the process of training and applying a system based on a given data set. It equips a system in such a way that it is able to learn and adapt without being explicitly programmed [34–36]. There exist a vast number of methods under the umbrella of machine learning, of which neural networks have been focused on in this paper. Artificial neural network is a type of network that is inspired by the biological neural network of the human brain [37,38]. Neural networks have been applied to a variety of fields [39,40]. The present work is based on the application of neural networks in cryptography, which is popularly termed as neural cryptography [41–43].

Kosko [44] proposed the introduction of bidirectionality in neural nets in order to store two-way memory associations. The two directions of information are stored in matrices W and W^T respectively. This is known as bidirectional associative memory (BAM). BAM exhibits the behavior of heteroassociative content addressable memory (CAM). A neural network based on BAM is known as a BAM neural network (BAMNN). Like training any neural network, this type of network is trained on the basis of inputs and targets. In this case, the BAMNN has been trained to simulate random binary noise so that key generation can be carried out in a random fashion.

4. Proposed methodology

The proposed bio-inspired cryptosystem comprises of an encryption algorithm, a key generation algorithm and a decryption algorithm. The proposed algorithms are discussed below.

4.1. Encryption algorithm

The encryption algorithm comprises the following components.

(i) Conversion of Genetic Code : This is the process of converting the binary form to DNA in Genetic Code.

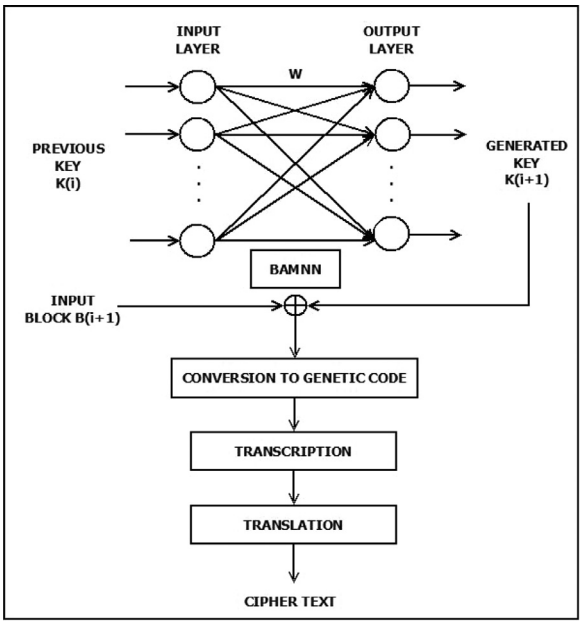

Fig. 2. Process of encryption and key generation in the proposed cryptosystem.

(ii) Transcription : This is the process of converting the DNA to mRNA form.
(iii) Translation : This is the process of converting the mRNA form to cipher text containing protein bases.

The encryption algorithm involves converting the readable text of the sender into the cipher text which is sent to the receiver. The readable text is converted to binary using Huffman encoding [31]. The string is then padded with k zeroes to make it a multiple of 16. This is to facilitate XOR with the 16-bit key in each round. The binary form obtained is XORed with the key and converted to the genetic code of DNA. Translation is then carried out to convert the DNA into mRNA. The mRNA is padded with '*N*' or '*NN*' in order to make it in 3-mRNA form. This is followed by translation in order to convert the mRNA to protein form, which yields the cipher text. All of the above processes were simulated using Python for the proposed cryptosystem.

4.2. Key generation algorithm

A Bi-directional Associative Memory Neural Network (BAMNN) is trained to generate the keys for every corresponding block of input. The first block is generated randomly and further blocks are generated using the BAMNN process. The architecture of the encryption and key generation algorithms in the proposed cryptosystem is depicted in Fig. 2. The key generation process is mainly carried out using a neural network with the property of bidirectional associative memory. The weight matrix W is used to represent the weights of the neural network. In this case, it is a 16×16 matrix of 256 weights. These weights are trained over a number of iterations using random binary noise. The final matrix is used for the purpose of key generation. Since the neural network uses both its inputs and outputs as keys, the number of input and output neurons is required to be equal.

4.3. Decryption algorithm

The decryption algorithm comprises the following components.

(i) Reverse Translation : This is the process of converting the cipher text containing protein bases to mRNA form.
(ii) Reverse Transcription : This is the process of converting the mRNA to DNA form.

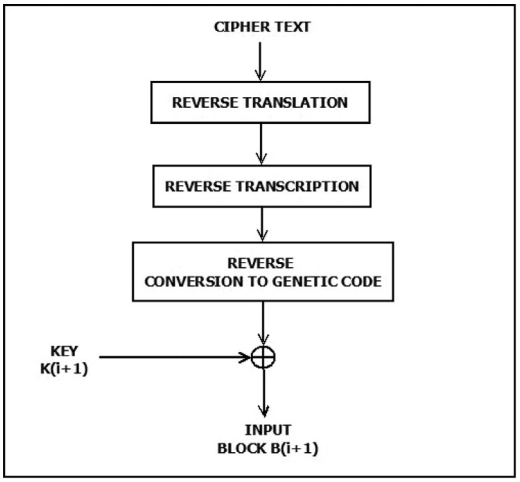

Fig. 3. Process of decryption in the proposed cryptosystem.

(iii) Reverse Conversion of Genetic Code : This is the process of converting the DNA in Genetic Code to Binary form.

The above processes in decryption have been simulated using Python. A significant challenge was in the case of reverse translation where a protein base and a 3-mRNA have a one-to-many mapping. This problem was solved using hashing. This enabled the conversion from protein base to mRNA form. After this process, the padded '*N*' or '*NN*' string at the end of the resulting mRNA was removed. The reverse transcription process has been simulated by converting the uracil base into thymine. This enabled the conversion from mRNA to DNA form. After converting the DNA form to binary, reverse conversion of genetic code was simulated by mapping the DNA to their corresponding binary form. The key for the particular round is then XORed with the binary form and the parameter k was used to remove padded zeroes and Huffman decoding [31] was performed to get the original decrypted plaintext. Since certain processes involved in the decryption algorithm may not be feasible in biology, they have been computer simulated for the purpose of the proposed cryptosystem.

The architecture of the decryption algorithm in the proposed cryptosystem is shown in Fig. 3.

5. Results and discussion

5.1. System configuration

The proposed cryptosystem was simulated using Python 3.5 with BioPython [45] on a system with 2.4 GHz processor speed and 4 GB RAM.

5.2. Key generation using BAMNN

A Bidirectional Associative Memory Neural Network is trained to simulate binary noise. The binary string consisting of 0s and 1s is converted to bipolar form (−1 and 1) to facilitate training of the activation functions of the neural network. The neural network is trained for 50 epochs for 16-bit random inputs and targets to simulate random binary noise. The 16 × 16 matrix of weights in each individual epoch is stored in w_i. This weight matrix is updated for each of the 50 epochs using Eq. (1). Finally, the trained weight matrix W over 50 epochs is obtained using Eq. (2).

$$w_i = x_i^T . y_i \tag{1}$$

$$W = \sum w_i \tag{2}$$

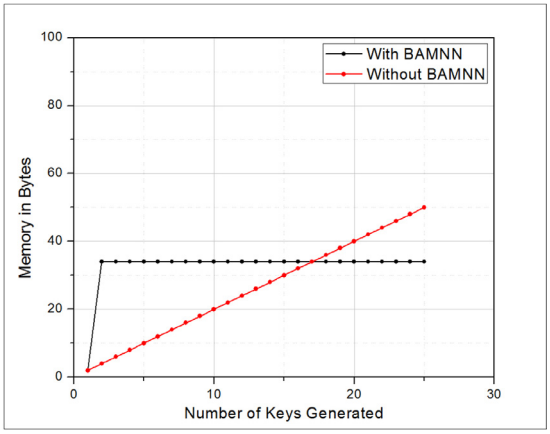

Fig. 4. Memory in bytes vs. number of keys generated.

The first key $K(0)$ is always a generated random 16-bit string. Subsequent keys are generated using the weight matrix W using Eq. (3).

$$K' = K(i).W \tag{3}$$

The key for the next round is finally obtained using the function in Eq. (4) where K'_j denotes an element of K'.

$$K(i+1) = \begin{cases} -1, & \text{if } K'_j < 0 \\ +1, & \text{if } K'_j > 0 \\ K(i), & \text{if } K'_j = 0 \end{cases} \tag{4}$$

The generated keys are re-converted back to the binary form before applying them to the encryption and decryption algorithms. The transpose of W, denoted by W^T, is used to compute the keys in the reverse order during the process of decryption. The calculation is carried out in reverse order using the last key value $K(n)$ as shown in Eqs. (5) and (6).

$$K'' = K(i).W^T \tag{5}$$

$$K(i-1) = \begin{cases} -1, & \text{if } K''_j < 0 \\ +1, & \text{if } K''_j > 0 \\ K(i), & \text{if } K''_j = 0 \end{cases} \tag{6}$$

This system particularly saves memory space even in case of larger sizes of data. In conventional algorithms, the entire generated key would have to be stored in memory to facilitate encryption and decryption. However, in the proposed algorithm, the memory storage is significantly reduced with increase in the size of the input data (as the number of keys generated depends on the size of the plaintext) as shown in Fig. 3. For the proposed key generation algorithm, only the first key (for encryption) or the last key (for decryption) needs to be stored while the other keys can be remembered and regenerated using the BAMNN. If only 1 key needs to be generated, 2 bytes (16 bits) are necessary while for every additional key generated, in total only 34 bytes (=2 + 32 bytes) are needed where 2 bytes is the size of the first key or last key and 32 bytes is the size of the weight matrix of the trained network.

From Fig. 4, it is evident that initially, key generation with BAMNN may prove costly in terms of memory for lower number of bytes. However, for higher number of bytes, key generation with BAMNN is better in terms of memory due to its memory occupancy even with a greater number of keys being generated.

The randomness of the keys generated for each round is gauged using the binary entropy function in Eq. (7).

$$H(X) = -p_0 log_2 p_0 - p_1 log_2 p_1 \tag{7}$$

In the above equation, p_0 and p_1 represent $P(X = 0)$ and $P(X = 1)$ respectively. The average binary entropy of 0.9625 was obtained for

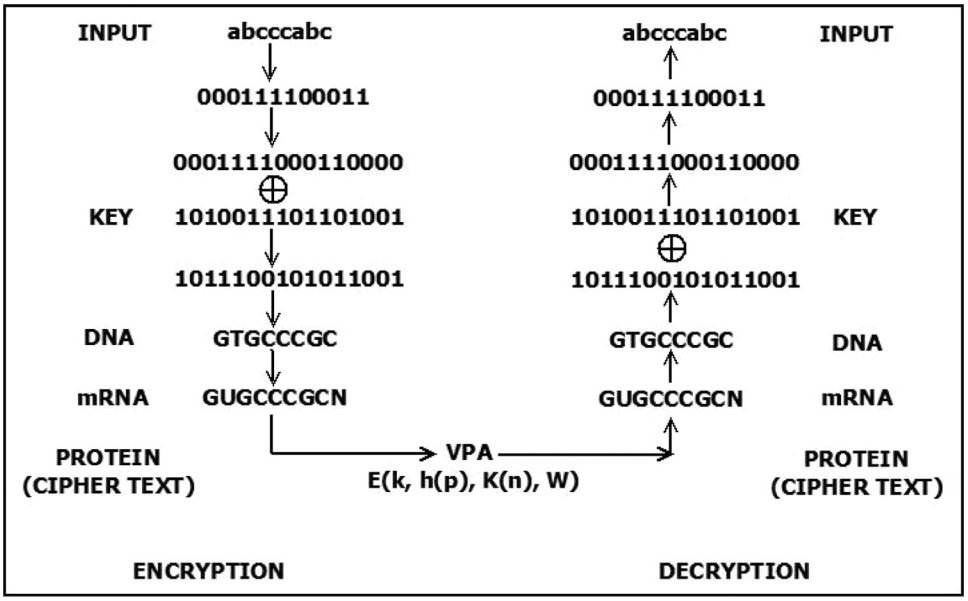

Fig. 5. Process of encryption and decryption in the proposed cryptosystem.

Table 1
Comparison of Encryption Time and Decryption Time of the Proposed Scheme with other schemes.

Kalsi et al. [24]			Proposed scheme	
Number of characters	Encryption time (ms)	Decryption time (ms)	Encryption time (ms)	Decryption time (ms)
500	0.00016401500	0.00026401450	40.01563504988	15.94296114843
1000	0.00035178900	0.00048178658	73.34018721002	28.60623965074
1500	0.00067315500	0.00077815800	124.17291612496	41.55519138396
Ning [25]			Proposed scheme	
Plaintext length (bits)	Encryption time (ms)	Decryption time (ms)	Encryption time (ms)	Decryption time (ms)
10(80)	244	423	8.79086666758	0.73770407674
100(800)	244	426	14.23020472677	2.94782272522
1000(8000)	290	462	73.34018721002	28.60623965074
10000(80000)	1285	1385	670.37585701974	325.9719732323

20 runs of the key generation algorithm. Increased randomness in the key makes it difficult for the attacker to guess the key. Moreover, the security will also increase with increase in the plaintext size as the key size also increases simultaneously.

5.3. Encryption and decryption using CDMB and DNA cryptography

For the purpose of encryption, the entered user text is converted to Huffman binary coding. Huffman coding also helps to obtain a dynamic length encoding of any entered string. For larger lengths of texts, it helps in optimum encoding compared to fixed-length binary encoding which makes the converted string extremely large. For instance, conversion of ASCII to 8-bit binary makes the converted text large for larger sizes of plaintext input. The input in this case is divided in the form of 16-bits and bitwise XOR is performed with the generated key (which is also generated as a string of 16-bits at a time). If any block is short of 16-bits, it is padded with k bits in order to make it 16 bits. The resultant is then altogether converted to genetic and DNA encoding using $00 \rightarrow A$, $01 \rightarrow C$, $10 \rightarrow G$ and $11 \rightarrow T$.

Next, the sequence is converted into mRNA and protein form using the processes of transcription and translation respectively. Forward translation is a one-to-one relationship from 3-mRNA (3 continuous non-overlapping strings of mRNA bases) to protein. However, backward translation is a one-to-many relationship from protein to 3-mRNA. This can be implemented using a hash which defines the relations from individual occurrences of proteins to 3-mRNAs. The translated protein may be directly transmitted or converted into binary form before transmission. It is also important to note that the encrypted forms of k value, the hashed protein values $h(p)$, final key value $K(n)$ and BAMNN weights W are also sent along with the ciphertext to the receiver.

Fig. 5 shows the entire process of encryption and decryption using a sample input. In this case, the input is lesser than or equal to 16 bits, and the process can be extended for inputs greater than 16 bits.

Table 1 shows a comparison of the encryption time and decryption time (in milliseconds) with that carried out in some previously proposed schemes. While the proposed scheme does not give sufficient performance compared to the scheme proposed in [24], it performs relatively better than the scheme proposed in [25] with regard to encryption and decryption times.

6. Performance analysis

6.1. Confusion and diffusion

Confusion and diffusion are security properties that were first discussed by Shannon [46]. Confusion means that each binary bit of the ciphertext should depend on several bits of the key. In other words, the relationship between the ciphertext and the key should be made as intricate as possible. In the proposed cryptosystem the ciphertext, when converted to binary, depends on more than one bit of the key. This property is also facilitated by the CDMB to a certain extent. The process of translation is a one-to-many relationship, which means a particular protein can be produced as a result of combinations of 3 mRNA bases at a time (referred to as 3-mRNA). The plaintext or key cannot

Table 2
Comparison of Proposed Scheme with others with regard to various attacks.

	Brute-force attack (including key guessing)	Ciphertext-only attack	Known plaintext attack	Differential cryptanalysis attack
[22]	√	√	√	–
[27]	√	–	–	✗
[28]	√	–	–	–
Proposed scheme	√	√	√	√

be independently used to derive the ciphertext as several operations are performed to combine both to produce the result. Diffusion means that if one character of the plaintext is changed, then several bits of the ciphertext get changed. This is achieved using the proposed cryptosystem as the processes of transcription and translation will produce changes in the ciphertext when a single bit of the plaintext is changed. It is also important to note that the same plaintext will generate different ciphertexts using the proposed cryptosystem since the key generated will be different every time. Hence the proposed scheme achieves the properties of confusion and diffusion.

6.2. Avalanche effect

Avalanche Effect is a cryptographic property where a slight change in the plaintext results in a significant change in the ciphertext.

For the proposed cryptosystem, change of 1 bit in the plaintext (in binary form) randomly results in changes of 93.59% bits of the ciphertext on average, while changing 2 bits in the plaintext (in binary form) randomly results in changes of 96.15% bits of the ciphertext on average. This shows that the proposed cryptosystem has good randomization. It also provides a difficult and complex mapping between the plaintext and ciphertext.

6.3. Security against attacks

6.3.1. Brute-force attack

Brute-force attack is an attack where an attempt is made by the intruder to guess the keys using the possible solution space of the proposed system. Since the space occupied by the keys is of the order of multiples of 16-bit blocks, this forms a space of $n \times 2^{16}$ bits where n is the number of blocks. It is difficult to computationally calculate the keys. Moreover, even if one key is guessed, it is even more difficult to guess the weights of the 16×16 BAMNN matrix which is used to generate the other keys. Further, it is computationally difficult to map all possibilities of proteins to their corresponding 3-mRNA codons without any specified hashed protein values. Hence, the proposed cryptosystem is resilient to the brute-force attack.

6.3.2. Ciphertext-only attack

In ciphertext-only attack, the attacker has access to a set of ciphertexts as well as certain information about the plaintext such as the language or statistical distribution of the plaintext characters. The attack is deemed successful if the attacker can deduce the plaintext and possibly the key. This will not work in the proposed system due to the high entropy of the generated keys with which the plaintext is XORed. Further operations lead to higher obscurity of the plaintext. In other words, the most frequently occurring plaintext character will not map to the most frequently occurring ciphertext character. The relationship between plaintext and ciphertext becomes even more intricate due to different sizes of plaintext and ciphertext (as observed in all simulated test cases). Hence, the proposed cryptosystem is able to withstand the ciphertext only attack.

6.3.3. Known plaintext attack

In known plaintext attack, the attacker tries to find the key by analyzing the associations between the sets of the plaintext and ciphertext. Firstly, it is difficult to decipher the key as DNA computing contains several large-scale computations. In this context, even if the attacker recovers the key from the sets, a different set of keys is generated for every execution freshly. Hence, the proposed cryptosystem can withstand the known plaintext attack.

6.3.4. Differential cryptanalysis attack

Differential cryptanalysis attack is a chosen plaintext attack where the attacker derives ciphertexts from a set of chosen plaintexts. This attack also fails because of fresh generation of keys which results in different ciphertexts each time. Table 2 summarizes the performance of the proposed scheme with other schemes with regard to various attacks. In the table, √ represents prevention of a particular attack while ✗ represents failure of withstanding a particular attack.

6.4. Achievement of CIA

6.4.1. Confidentiality

Confidentiality involves maintenance of secrecy from unauthorized entities while exchanging information. This has been established in the proposed system as all the transmitted entities and parameters are encrypted.

6.4.2. Integrity

Data integrity ensures that the recipient does not receive any manipulated information as a result of insertion, deletion and modifications. In other words, even if the information is manipulated, the recipient should have some form of mechanism to ensure that the information received has been changed. Data integrity has been achieved in this scheme by using hashing. The hashed protein values will differ between the sender and the receiver if the intruder maliciously changes the data before the receiver receives it.

6.4.3. Availability

Availability involves guaranteeing access of information to people whenever required. The proposed cryptosystem is a simulated one and is therefore available at all times. It also supports large forms of text input for encryption and decryption without loss of information, that is, it is a lossless scheme where no bits are lost during transmission. The proposed cryptosystem algorithm has also been tested using whitespaces and special characters for sizes up to 10,000 characters of plaintext.

7. Conclusion and future work

A Bio-Inspired Cryptosystem has been proposed that uses Machine Learning and DNA Cryptography for securing information. In this system, the text is first encoded in 16-bit blocks each using probability-based variable-length Huffman encoding. Key generation has then been carried out block-wise to simulate random binary noise using BAMNN where the average binary entropy of 0.9625 has been obtained. BAMNN also serves to considerably save on memory requirements compared to key generation systems that do not use BAMNN. The encoded sequence is then XORed with the generated keys and the entire result is genetically encoded. Encryption and decryption have been carried out using DNA Cryptography and CDMB respectively. Avalanche effect shows that random change of 1 bit in the plaintext results in 93.59% change in bits of the ciphertext on average, while random change of 2 bits in the plaintext results in 96.15% change in bits of the ciphertext on average. The

proposed cryptosystem has been proven to be secure to the brute force, ciphertext only, known plaintext and differential cryptanalysis attacks. It has been tested on a variety of data involving whitespaces and special characters. It is also able to perform on big data tested with sizes up to 10,000 characters. The performance of the proposed scheme has been compared to that of others with regard to resistance of attacks. The system has competent encryption and decryption times even on large data sizes when compared with previous schemes. The proposed cryptosystem also achieves the CIA principle.

Future work in this direction includes incorporating additional security and authentication measures to strengthen this scheme. Artificial Intelligence also has enormous scope of application in the area of cryptography. DNA cryptography is still in a nascent state and more powerful cryptosystems can be built using this computing paradigm.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.sysarc.2019.02.005.

References

[1] W. Stallings, Cryptography and Network Security: Principles and Practice, Pearson Education India, 2003.
[2] S.S. Greene, Security policies and procedures, Principles and practices. Upper Saddle, 2006.
[3] M. Karuppiah, R. Saravanan, A secure remote user mutual authentication scheme using smart cards, J. Inf. Secur. Appl. 19 (4–5) (2014) 282–294.
[4] M. Karuppiah, R. Saravanan, A secure authentication scheme with user anonymity for roaming service in global mobility networks, Wirel. Pers. Commun. 84 (3) (2015) 2055–2078.
[5] M. Karuppiah, S. Kumari, X. Li, F. Wu, A.K. Das, M.K. Khan, R. Saravanan, S. Basu, A dynamic id-based generic framework for anonymous authentication scheme for roaming service in global mobility networks, Wirel. Pers. Commun. 93 (2) (2017) 383–407.
[6] S. Kumari, M. Karuppiah, X. Li, F. Wu, A.K. Das, V. Odelu, An enhanced and secure trust-extended authentication mechanism for vehicular ad-hoc networks, Secur. Commun. Netw. 9 (17) (2016) 4255–4271.
[7] M. Karuppiah, R. Saravanan, Cryptanalysis and an improvement of new remote mutual authentication scheme using smart cards, J. Discrete Math. Sci. Cryptogr. 18 (5) (2015) 623–649.
[8] M. Wazid, A.K. Das, R. Hussain, G. Succi, J.J. Rodrigues, Authentication in cloud–driven IoT-based big data environment: survey and outlook, J. Syst. Archit. (2018).
[9] G.P. Saggese, L. Romano, N. Mazzocca, A. Mazzeo, A tamper resistant hardware accelerator for RSAcryptographic applications, J. Syst. Archit. 50 (12) (2004) 711–727.
[10] G.J. Simmons, Symmetric and asymmetric encryption, ACM Comput. Surv. (CSUR) 11 (4) (1979) 305–330.
[11] J. Thakur, N. Kumar, DES, AES and blowfish: symmetric key cryptography algorithms simulation based performance analysis, Int. J. Emerging Technol. Adv. Eng. 1 (2) (2011) 6–12.
[12] H.-C. Chen, J.-C. Yen, A new cryptography system and its VLS realization, J. Syst. Archit. 49 (7–9) (2003) 355–367.
[13] G. Cui, Y. Liu, X. Zhang, New direction of data storage: DNA molecular storage technology, Comput. Eng. Appl. 42 (26) (2006) 29–32.
[14] H. Feistel, Cryptography and computer privacy, Sci. Am. 228 (5) (1973) 15–23.
[15] P. Mahajan, A. Sachdeva, A study of encryption algorithms AES, DES and RSA for security, Global J. Comput. Sci. Technol. (2013).
[16] H. Handschuh, L.R. Knudsen, M.J. Robshaw, Analysis of SHA-1 in encryption mode, in: Cryptographers' Track at the RSA Conference, Springer, 2001, pp. 70–83.
[17] A. Roy, A. Nath, DNA encryption algorithms: scope and challenges in symmetric key cryptography, Int. J. Innovative Res. Adva. Eng. 3 (11) (2016).
[18] A. Cherian, S.R. Raj, A. Abraham, A survey on different DNA cryptographic methods, Int. J. Sci. Res. 2 (4) (2013) 167–169.
[19] J. Blackledge, S. Bezobrazov, P. Tobin, Cryptography using artificial intelligence, in: Neural Networks (IJCNN), 2015 International Joint Conference on, IEEE, 2015, pp. 1–6.
[20] A. El-Zoghabi, A.H. Yassin, H.H. Hussien, Survey report on cryptography based on neural network, Int. J. Emerging Technol. Adv. Eng. 3 (12) (2013) 456–462.
[21] P. Xie, M. Bilenko, T. Finley, R. Gilad-Bachrach, K. Lauter, M. Naehrig, Crypto-nets: neural networks over encrypted data (2014) arXiv:1412.6181.
[22] R. Ramamurthy, C. Bauckhage, K. Buza, S. Wrobel, Using echo state networks for cryptography, in: International Conference on Artificial Neural Networks, Springer, 2017, pp. 663–671.
[23] V. Sagar, K. Kumar, A symmetric key cryptography using genetic algorithm and error back propagation neural network, in: Computing for Sustainable Global Development (INDIACom), 2015 2nd International Conference on, IEEE, 2015, pp. 1386–1391.
[24] S. Kalsi, H. Kaur, V. Chang, DNA cryptography and deep learning using genetic algorithm with NW algorithm for key generation, J. Med. Syst. 42 (1) (2018) 17.
[25] K. Ning, A pseudo dna cryptography method (2009) arXiv:0903.2693.
[26] M. Lu, X. Lai, G. Xiao, L. Qin, Symmetric-key cryptosystem with DNA technology, Sci. China Ser. F 50 (3) (2007) 324–333.
[27] U.N. Hussain, T. Chithralekha, A.N. Raj, G. Sathish, A. Dharani, A hybrid DNA algorithm for des using central dogma of molecular biology (CDMB), Int. J. Comput. Appl. 42 (20) (2012) 1–4.
[28] R. Jogdand, S.S. Bisalapur, Design of an efficient neural key generation, Int. J. Artif. Intell. Appl. 2 (1) (2011) 60–69.
[29] J. Mandal, D. Datta, A. Sarkar, Hopfield network based neural key generation for wireless communication (HNBNKG), in: Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2014, Springer, 2015, pp. 217–224.
[30] L.M. Adleman, Molecular computation of solutions to combinatorial problems, Science 266 (5187) (1994) 1021–1024.
[31] D.A. Huffman, A method for the construction of minimum-redundancy codes, Proc. IRE 40 (9) (1952) 1098–1101.
[32] F. Crick, Central dogma of molecular biology, Nature 227 (5258) (1970) 561.
[33] J. Watson, F. Crick, A representation of the structure of dna, Bioscience 24 (11) (1974). 622–622
[34] N.M. Nasrabadi, Pattern recognition and machine learning, J. Electron. Imaging 16 (4) (2007) 049901.
[35] Z. Ghahramani, Probabilistic machine learning and artificial intelligence, Nature 521 (7553) (2015) 452.
[36] S.B. Kotsiantis, I. Zaharakis, P. Pintelas, Supervised machine learning: a review of classification techniques, Emerging Artif. Intell. Appl. Comput. Eng. 160 (2007) 3–24.
[37] R.J. Schalkoff, Artificial neural networks, 1, McGraw-Hill New York, 1997.
[38] M.T. Hagan, H.B. Demuth, M.H. Beale, et al., Neural network design, 20, Pws Pub. Boston, 1996.
[39] P.K. Simpson, Neural networks applications, IEEE Press, 1997.
[40] S.B. Wankhede, Analytical study of neural network techniques: SOM, MLP and classifier-a survey, IOSR J. Comput. Eng. (IOSR-JCE) 16 (3) (2014) 86–92.
[41] W. Kinzel, I. Kanter, Neural cryptography, in: Neural Information Processing, 2002. ICONIP'02. Proceedings of the 9th International Conference on, 3, IEEE, 2002, pp. 1351–1354.
[42] E. Volna, M. Kotyrba, V. Kocian, M. Janosek, Cryptography based on neural network., in: ECMS, 2012, pp. 386–391.
[43] R. Mislovaty, E. Klein, I. Kanter, W. Kinzel, Security of neural cryptography, in: Electronics, Circuits and Systems, 2004. ICECS 2004. Proceedings of the 2004 11th IEEE International Conference on, IEEE, 2004, pp. 219–221.
[44] B. Kosko, Bidirectional associative memories, IEEE Trans. Syst. Man Cybern. 18 (1) (1988) 49–60.
[45] P.J. Cock, T. Antao, J.T. Chang, B.A. Chapman, C.J. Cox, A. Dalke, I. Friedberg, T. Hamelryck, F. Kauff, B. Wilczynski, et al., Biopython: freely available python tools for computational molecular biology and bioinformatics, Bioinformatics 25 (11) (2009) 1422–1423.
[46] C.E. Shannon, A mathematical theory of cryptography, Math. Theory Cryptogr. (1945).

Sayantani Basu is with the School of Computing Science and Engineering, VIT University, Vellore, India. With a deep interest for research in the field of network security, she has executed few projects and authored a few publications in this area.

Marimuthu Karuppiah received his B.E. degree in computer science and engineering from Madurai Kamaraj University, Madurai, India in 2003, M.E. degree in computer science and engineering from Anna University, Chennai, India in 2005, Ph.D. degree in computer science and engineering from VIT University, Vellore, India in 2015. He is now an associate professor in School of Computing Science and Engineering, VIT University, Vellore, India. He has published more than 20 research papers in SCI indexed journals. Also, he has published more than 35 research papers in SCOPUS indexed journals and international conferences. He is a life member of Cryptology Research Society of India (CRSI) and Computer Society of India (CSI). Also, he is a senior member of IEEE and member of ACM. His main research interests include cryptography and wireless network security, in particular, authentication and encryption schemes.

Mita Nasipuri received her B.E.Tel.E., M.E.Tel.E, and Ph.D. (Engg.) degrees from Jadavpur University, in 1979, 1981 and 1990, respectively. Prof. Nasipuri has been a faculty member of J.U since 1987. Her current research interest includes bioinformatics, image processing, pattern recognition, and multimedia systems. She is a senior member of the IEEE, U.S.A., Fellow of I.E. (India) and W.B.A.S.T, Kolkata, India.

Anup Kumar Halder is a Vivesvaraya Ph.D. scholar in the Department of Department of Computer Science and Engineering, Jadavpur University, Kolkata, India. He received his B.Tech degree in computer science and engineering from Bankura Unnayani Institute of Engineering, Bankura, India in 2011 and M.E degree in Computer Science and Engineering from Jadavpur University, Kolkata, India in 2014. His area of current research interest is bioinformatics.

Niranchana Radhakrishnan received her B.E. degree in computer science and engineering from Anna University, Chennai, India in 2005, M.E. degree in computer science and engineering from Anna University, Chennai, India in 2008. In 2009, she was appointed as a lecturer in Department of Computer Science and Engineering at Ganathipathy Tulsis Jain Engineering College, Vellore, India. Now she is pursuing her Ph.D. degree at VIT University, Vellore, India. She is a student member of IEEE. Her main research interests include cryptography and wireless network security, wireless sensor networks in particular, authentication and encryption schemes.